In the Claims

1. (Currently Amended) An article comprising a machine readable medium storing instructions that, if executed by a machine, cause the machine to perform a plurality of operations comprising:

    specifying a monitor address;

    suspending a thread until a monitor break event occurs;

    testing whether the monitor break event is a write to a first value for a first task indicated by the monitor address;

    testing whether the monitor break event is a write to a second value for a second task indicated by the monitor address;

    if the monitor break event is not ~~the write to the monitor address~~ a write to the first value or a write to the second value, then suspending the thread again.

2. (Previously Presented) The article of claim 1 wherein suspending the thread again comprises returning to specifying the monitor address.

3. (Previously Presented) The article of claim 2 wherein specifying the monitor address comprises executing a MONITOR instruction and wherein suspending the thread until the monitor break event occurs comprises executing an MWAIT instruction.

4. (Currently Amended) The article of claim 1 wherein said plurality of operations

further comprise, after specifying the monitor address and before suspending the thread:

testing whether ~~data at the monitor address~~the first value or the second value has changed.

5. (Currently Amended) The article of claim 1 wherein specifying the monitor address comprises executing an instruction with an implicit operand ~~chosen from a set consisting of~~ specifying a linear address~~, a virtual address, a physical address, and a relative address~~.

6. (Currently Amended) The article of ~~claim 5~~ claim 1 wherein the operand is ~~one of a second set consisting of an explicit operand and~~ an implicit operand.

7. (Currently Amended) The article of claim 1 wherein said monitor address specifies a cache line and wherein said first value and said second value are respectively stored in a first and a second separate work location in the cache line.

8. (Previously Presented) The article of claim 2 wherein said plurality of operations further comprise providing a second operand as a mask operand to control which events are monitor break events.

9. (Previously Presented) An article comprising a machine readable medium storing instructions that, if executed by a machine, cause the machine to perform operations

comprising:

programming a monitor with a monitor address corresponding to a cache line of at least one work location;

suspending a thread until a monitor break event occurs;

testing whether the at least one work location indicates a first task is ready to execute;

testing whether the at least one work location indicates a second task is ready to execute;

if neither the first task nor the second task is ready to execute, then returning to suspending the thread.

10. (Previously Presented) The article of claim 9 wherein returning to suspending the thread until the monitor break event occurs further comprises re-programming the monitor with the monitor address prior to suspending the thread.

11. (Previously Presented) The article of claim 9 wherein returning to suspending the thread comprises returning to programming the monitor with the monitor address.

12. (Currently Amended) ~~A method comprising:~~ The method of claim 9 wherein said method further comprises, after programming the monitor and before suspending the thread:

testing whether data has changed for either of the first task or the second task at the at least one work location.

~~specifying a monitor address;~~

~~suspending a thread until a monitor break event occurs;~~

~~testing whether the monitor break event is a write to the monitor address;~~

~~if the monitor break event is the write to the monitor address, then suspending the thread again.~~

13. (Currently Amended) The method of claim 9 wherein programming the monitor comprises executing an instruction with an implicit operand. ~~12 wherein suspending the thread again comprises returning to specifying the monitor address.~~

14. (Currently Amended) The method of ~~claim 13~~ claim 9 wherein ~~specifying the monitor address~~ programming the monitor comprises executing a MONITOR instruction and wherein suspending the thread until the monitor break event occurs comprises executing an MWAIT instruction.

15. (Currently Amended) The method of ~~claim 12~~ claim 14 wherein said method further comprises, after ~~specifying the monitor address~~programming the monitor and before suspending the thread:

testing whether data at the monitor address has changed.

16. (Currently Amended) The method of ~~claim 12~~ claim 15 wherein specifying the monitor address comprises executing an instruction with ~~an operand chosen from a set consisting of~~ a linear address~~, a virtual address, a physical address, and a relative~~

~~address~~.

17. (Currently Amended) The method of claim 16 wherein ~~programming~~ the operand is ~~one of a second set consisting of an explicit operand and~~ an implicit operand.

18. (Currently Amended) The method of ~~claim 1~~ claim 17 wherein said method further comprises enabling recognition of writes to the monitor address as monitor break events.

19. (Previously Presented) The method of claim 13 further comprising providing a second operand as a mask operand to control which events are monitor break events.

20. (Currently Amended) A system comprising:

a processor;

a monitor to generate a monitor break event in response to a memory access to a monitor address;

event detect logic to detect an of a plurality of monitor break events;

a memory to store a loop in a first thread executable by said processor to specify said monitor address and to repeatedly suspend said first thread after monitor break events until the memory access to the monitor address occurs, the loop comprising:~~.~~

a test to determine whether a work location in a first cache line indicated by the monitor address contains a first value, wherein a first routine is executed if said work location contains the first value;

a second test to determine whether the work location in said first cache line contains a second value, wherein a second routine is executed if said work location contains the second value;

an instruction to suspend said first thread if said work location does not contain said first value and said work location does not contain said second value.

21. (Currently Amended) The system of claim 20 wherein said loop further comprises:

a first instruction to specify the monitor address;

a second instruction to suspend said first thread.

22. (Previously Presented) The system of claim 21 wherein said loop further comprises a test after said first instruction to determine whether data at the monitor address has changed after execution of the first instruction but before execution of the second instruction, wherein said loop exits without execution of the second instruction if data at the monitor address has changed.

23. (Previously Presented) The system of claim 21 wherein said loop further comprises a test after said first instruction to determine whether data at the monitor address has changed after execution of the second instruction wherein said loop performs another iteration if data at the monitor address has not changed.

24. (Canceled)

25. (Canceled)

26. (Canceled)

27. (Canceled)

28. (Canceled)

29. (Canceled)

30. (Canceled)

31. (New) The system of claim 20 wherein said monitor is programmable by a monitor instruction having an implicit operand.

32. (New) The system of claim 31 wherein said implicit operand is to specify a linear address.